IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 24, 2021, accepted October 23, 2021, date of publication November 8, 2021,
date of current version November 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3125962

Efficient Prediction Mode Decisions for Low Complexity MV-HEVC

SHAHID NAWAZ KHAN[1], KHURRAM KHAN[2], NAZEER MUHAMMAD[3], AND ZAHID MAHMOOD[1]
[1]Department of Electrical and Computer Engineering, COMSATS University Islamabad, Abbottabad Campus, Abbottabad 22060, Pakistan
[2]Department of Avionics Engineering, Air University, Islamabad 44230, Pakistan
[3]Department of IT and Computer Science, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Haripur 22620, Pakistan

Corresponding author: Zahid Mahmood (zahid0987@cuiatd.edu.pk)

ABSTRACT The increased number of possible inter-prediction blocks sizes, improved the coding efficiency of High Efficiency Video Coding (HEVC), but at the same time it also increased the complexity of the encoder. In this work, we propose an Efficient Inter-Prediction Mode Decision (EIPMD) technique for the Multi-View extension of the HEVC standard (MV-HEVC). We figure out the least and most probable inter-prediction Modes and use Coding Unit (CU) splitting information of the base view to find its relation with the prediction Modes. We improve the efficiency of our technique by optimizing the base view search area. In our technique, we also makes use of the current depth level of the CU and temporal level of the pictures. Simulation results show that compared to HTM [16.2], which is the reference encoder for MV-HEVC, our proposed technique reduces the encoding time by approximately 66% on average, with very minor change in other parameters.

INDEX TERMS Coding unit, inter-view, MV-HEVC, prediction unit, mode, video coding.

I. INTRODUCTION

With the development of fast Internet and mobile networks, now it is possible to send and receive large amount of data in much less amount of time. This eased the way to send and receive video data more frequently in recent times. Meanwhile, user data is shifting towards high resolution videos. By 2022, 20.3% of Internet video traffic would be Ultra HD and 56.2% would be HD [1]. This demand for high resolution videos will be a major factor in deciding the future networks and user end devices. High resolution videos generate huge amount of data. Standard compression methods are developed to compress the video data. The latest video coding standard is known as High Efficiency Video Coding (HEVC) [2]. The standard is developed by Joint Collaborative Team on Video Coding (JCT-VC) in 2013. The HEVC/ H.265 [3] is superior to its predecessor, 264/Advanced Video Coding (AVC) [4], by achieving twice the compression efficiency, whereas it maintains the same object quality [5]. In HEVC, a relatively large coding block with hierarchical block splitting and the increased number of Modes, results in the increase in computation complexity for inter and intra-prediction [6].

The associate editor coordinating the review of this manuscript and approving it for publication was Gangyi Jiang.

Stereoscopic and auto stereoscopic displays [7], [8] have opened the door to a new perception of virtual reality in the videos. Multi-View and 3D Videos are gaining more and more popularity. Multi-View Video Coding (MVC) [9] an extension of AVC was proposed to address multi-view videos, in which same scene is captured by multiple cameras aligned in a specific way. Multi-view videos are also used in virtual view synthesis [10] in the Free View Video (FVV) concept [11], [12]. The Multi-View extension of HEVC is known as MV-HEVC [13]. Due to the multiple views, the complexity of the encoder has also increased. To reduce the complexity of the encoder, researchers have proposed different techniques. Our work is also a part of this effort in which, we propose methods to reduce the complexity of the MV-HEVC encoder. Our contributions are listed below.

- We propose an Efficient Inter-Prediction Mode Decision (EIPMD) method, which utilizes the Coding Unit (CU) splitting information, of the base view. Based on the disparity between the views, neighborhood of the co-located Coding Tree Unit (CTU) of view-0 is utilized to set a threshold Mode level for the current Prediction Unit (PU) in dependent views.
- We further improve our proposed EIPMD technique by optimizing the decision window in the view-0. This makes our proposed method to be applicable on a

larger area of the picture. The encoder complexity is further reduced, without any significant loss in other parameters. We believe that our proposed method, EIPMD, is simpler and produces better results than the previously available methods in literature.

Rest of the paper is organized as follows. Section II gives an overview of the related works, Fundamentals of the CU splitting and PU mode decision are briefly discussed in Section III. Motivation behind the proposed methods is explained in Section IV. Section V explains the proposed EIPMD method. The evaluation parameters and Experimental setup, used in this work are discussed in Section VI. Section VII lists detailed results, analysis and comparisons. Finally, with some hints about possible future work, the paper is concluded in Section VIII

II. LITERATURE REVIEW

Due to the complex compression process adopted in HEVC, many researchers focused on finding ways to minimize the complexity of the encoder. One approach is to target the processes, which consume most of the encoding time. Another approach is to target the processes which have become more complex in this standard as compared with the previous one. As Intra-prediction process in HEVC is more complex than the AVC, therefore, many researchers [14], [15] targeted the intra prediction for complexity reduction of HEVC encoder. Different correlations and parameters are exploited to restrict the intra-prediction process to minimize the complexity of the encoder.

The complex Quad tree CU splitting process of HEVC is targeted by most of the researchers, in which the aim is to predict decision earlier or restrict the CU splitting process from checking all CU sizes. We introduce the reader to some of good research work found in literature in this domain. Choi *et al*. [16] proposed an Early CU decision algorithm, which reduces the CU splitting process considerably. A lot of research work is available in the literature, which is targeting the early detection of Skip Mode. A simple and effective method to detect Early Skip Detection (ESD) is proposed by Won *et al*. [17]. Another fast Mode decision technique, based on Coded Block Flag (CBF) is proposed by Gweon and Lee [18]. These methods are also incorporated in the reference encoder and can be enabled from the configuration file. A fast CU decision method is proposed in [19], which is based on the observation that the CU depth level is content dependent. The method makes use of previously encoded frame and neighborhood to decide the CU depth range. Based on Bayesian decision rule a fast early CU termination method is proposed in [20], that is uses the online and offline training. A pyramid motion divergence based fast CU decision method is proposed in [21] that makes use of motion vector distribution. A fast CU decision method is proposed in [22], which makes use of CBF and RD cost to skip PUs. Variance of absolute difference is proportional with RD cost, on the basis of this observance a Markov random field based Fast CU decision method is proposed in [23]. Based on Rate Distortion (RD) cost, Skip Mode and CU splitting decisions are made in [24] to reduce the CU splitting complexity. A method based of latent sum of absolute difference estimation is proposed in [25], in which two-layer motion estimation is designed. Motion compensated exponent model of RD and (Sum of Absolute Difference) SAD cost is used for fast CU decision.

HEVC uses a more complex Prediction Modes structure as compared with its predecessor, so this also attracted many researchers. In this domain, the researchers tried to reduce the Modes decision process by predicting the most probable Modes or by restricting the Modes selection process to certain Modes. Rhee *et al*. in [26] discussed fast Mode decision methods proposed for the previous standard and examines their effectiveness on HEVC. In [27], Shen *et al*. proposed three adaptive Mode decision techniques using the inter-level and quad-tree correlation. Lee *et al*. [28], proposed an early Skip decision technique. The method utilizes the distortion characteristics to decide the SKIP Mode, after calculating RD cost for 2N $\times$ 2N Merge Mode. A fast inter-prediction Mode decision method is proposed in [29]. This method utilizes the edge information to reduce the complexity of RDO process. Utilizing the motion estimation and hierarchical depth correlation for HEVC, an Early Merge Mode decision technique is proposed in [30]. An early Merge Mode decision method for dependent texture views is proposed in [31]. Inter-view and depth correlation of coding Modes are utilized to reduce the complexity by terminating the unwanted prediction Modes. An early Direct Mode decision method for low complexity MVC is proposed in [32]. It utilizes the property of all zero block, RD cost and the correlation of current Macro-blocks with spatio-temporal nearby Macro-blocks. This method targets the previous standard. Tohidypour *et al*. [33] proposed an adaptive fast Mode assigning method based on Bayes classifier to predict the perdition-Mode using neighboring CUs.

The works mentioned above, target different parts of the encoder for complexity reduction. Some target the Intra-prediction, some target the Quad-tree CU splitting process, and some target the inter-prediction Modes. We in this work focus on reducing the complexity of the MV-HEVC encoder by optimizing the inter-prediction Mode decision. A fast Mode decision technique, based on adaptive ordering of Modes is proposed in [34], which uses RD cost and bit cost statistics. Recently in [35] a fast Mode decision algorithm for MV-HEVC is proposed, which is based on perceptual distortion model. It uses 3D Sobel-model and the Binocular Just Noticeable Distortion (BJND) to propose Perceptual Distortion Threshold model (PDTM) for Multi-View video coding. The algorithm estimates the Sum of Squared Errors (SSE) of the current CU, and on the basis of a threshold, it terminates the unnecessary Modes to minimize the complexity of the encoder.

The aforementioned work is a nice addition in low complexity MV-HEVC techniques. However, we discover, that the Mode selection process of MV-HEVC can be further

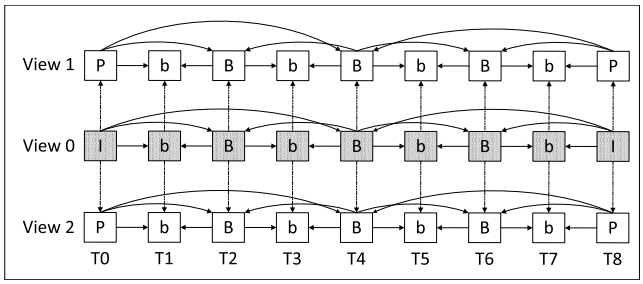


FIGURE 1. Typical hierarchical-B coding structure for 3-view MV-HEVC, used in this work. View-0 is the independent view/ base view, whereas views-1/2 are dependent views.

optimized in a much simpler way, if some other parameters are also explored. Therefore, in this work we explore these parameters and on the basis of that, we propose a much simpler technique. Our proposed technique is based on a novel idea of utilizing the correlation between the CU splitting of base view with the PU decisions of the dependent views. To the best of our knowledge, utilizing the CU splitting information of the base view for the PU decisions of the dependent views, is not reported in the literature earlier. In addition to this novelty, our proposed technique produces better results than the methods available in literature for low complexity MV-HEVC. For the readers, to get familiar with the terminology, we briefly introduce some basics in the next section.

III. TECHNICAL BACKGROUND

Prediction structure for 3–view MV-HEVC is shown in Fig. 1. First the view–0 i.e., the base view is encoded independently, hence also known as the independent view. Then view–1 and view–2 are encoded which also use reference pictures of the view–0, hence are called dependent views. The square boxes shown in figure represent the pictures of the video. In this case, $T0$–$T8$ represent the pictures of the video. Arrows are used to show the prediction used by each picture. As an example the picture $T1$ in view–2 is getting reference from $T0$, $T2$, and also $T1$ of the view–0. Apart from these immediate reference pictures, it can also use previously encoded reference pictures for reference, which are available in reference picture list. The inter-view prediction mechanism improves the coding efficiency of MV-HEVC, but this this comes at a cost of increase in the encoding complexity [13], [36]. Apart from the minor changes for inter-view prediction, the HEVC uses the same Quad tree partitioning structure [37]–[40] which is used by the HEVC. In HEVC, the picture is divided into 64 × 64 square blocks called CTUs. This size can be adjusted, within specified range, using the configuration file. Fig. 2 (a) shows the picture of size 1024 × 768 partitioned into 12 rows with each row containing 16 CTUs. Fig. 2 (b) expands a single CTU and shows the processing order inside a CTU. In Fig. 2 (c) the splitting of the CTU in CUs and their corresponding depth levels is shown. The CUs can be either inter-predicted (i.e., from previously encoded pictures) or intra predicted (i.e., from the same picture that is currently being encoded). Each CU of size 2N × 2N can be divided into one, two or four PUs, depending upon prediction Mode, and depth level of the CU. Fig. 5, shows the possible inter-prediction PUs for all possible depth levels of the CUs. Intra-prediction only uses sizes 2N × 2N and N × N for prediction. Fig. 3 shows the CU splitting and PU selection process of the HEVC encoder, in which the ESD, ECU and the CBF-Fast methods are also shown. These methods are used only if activated in the configuration file. Rest of the CU splitting and prediction Mode selection process is shown in the Fig. 3. Asymmetric Motion Partition (AMP) Mode is evaluated according to the Table. 1. The complexity of the process can also be noticed. As investigated by [42], the encoding time of HEVC is higher than H.264/AVC by 253% on average, making it impractical for implementation in multimedia applications. Therefore, it is necessary to significantly reduce the encoding complexity of HEVC with negligible loss in Rate-Distortion (RD) performance.

A high proportion of the encoding time is however devoted to inter-prediction during the encoding phase. Inter frame prediction alone consumes from 60% to 96% of the total encoding time [43]. In [44], inter prediction is shown to consume 84% of the encoding time while only 3% is taken by intra-prediction for High Definition (HD) sequences.

To reduce the complexity of the encoder many researchers focused on reducing the complexity of this process in order to reduce the overall complexity of the encoder. According to findings of the [30], on average 88% of the times the Merge/Skip Mode is selected. Therefore, researchers focused on finding out a mechanism to detect these Modes earlier to reduce the complexity of the encoder. A new technique included in the HEVC is the Merge Mode. It uses Motion Vectors (MVs) of seven candidates to get the motion information that can be used for the current PU. As shown in the Fig. 2 (d), it uses five spatial and two temporal PU candidate. Motion vectors of these candidates are used to calculate the RD cost and the one with minimum cost is used for the current PU [45]. This reduces the computational complexity of the motion estimation process because only the index of the best candidate is transmitted. For a PU, if there exists such a PU in its Merge candidate, who has no residual after transform and quantization, then its coded with Skip Mode. Prediction residual is not transmitted in Skip Mode. Hence RD cost of Skip Mode is less than other Modes.

Video consists of pictures cascaded in series, where each picture can be decoded independently. Fig. 4 shows a Group Of Pictures (GOP) of eight pictures. GOP length can be adjusted using configuration file within defined limits. The sequence in which the pictures are captured is recognized by Picture Order Count (POC). The order in which the pictures are encoded are known as Encoding Order (EO). EO is decided to maximize the coding efficiency, because previously encoded pictures can be used as reference pictures for current picture being encoded. Fig. 4 shows the POC, with EO and also the temporal levels of the pictures. One

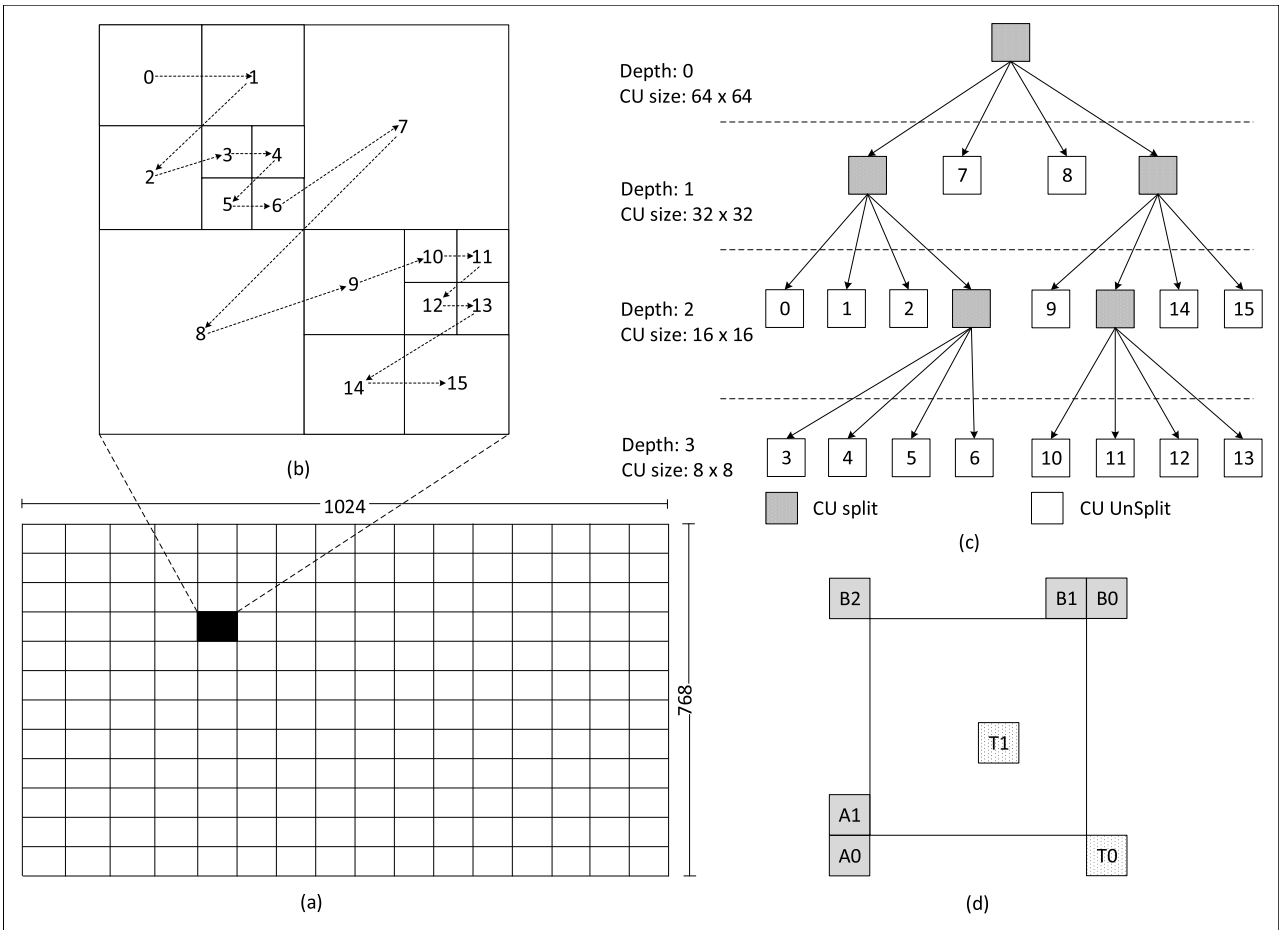


FIGURE 2. A 1024 × 768 HEVC frame consisting of Coding Tree Units (CTU). The CTU partitioning into coding units (CUs) along with processing order and depth information. (a) HEVC frame, (b) A CTU with possible CU splitting, and (c) corresponding CUs with depth and size information.

TABLE 1. Fast asymmetric motion partition (AMP) mode evaluation [41].

Conditions	Actions
Best Mode is $(2N \times N)$	Try $2N \times nU$ and $2N \times nD$
Best Mode is $(N \times 2N)$	Try $nL \times 2N$ and $nR \times 2N$
Best Mode is $((2N \times 2N)$ && (!Merge Mode)&& (!Skip Mode))	Try all AMP Modes
Parent CU is AMP Mode	Only try Merge Mode for all AMP Modes
Parent CU is $((2N \times 2N)$ && (!skipped))	Only try Merge Mode for all AMP Modes
Parent CU is ((intra) && (best Mode is $2N \times N$))	Only try Merge Mode for $2N \times nU$ and $2N \times nD$
Parent CU is ((intra) && (best Mode is $N \times 2N$))	Only try Merge Mode $nL \times 2N$ and $nR \times 2N$
CU size is 64×64	No AMP Modes are evaluated

way to look at the temporal levels is the Δ POC (i.e., the temporal distance between the current and reference picture), value. At temporal level-1 Δ POC is 8, which means that the available reference picture for a picture at temporal level-1 is eight pictures away from picture which is currently being encoded. Similarly for pictures in temporal level-4 Δ POC value is 1, which means that at temporal level-1, adjacent pictures are available as reference pictures for prediction. The arrows in upper part of the Fig. 4 show the prediction structure. At temporal level-4 there is minimum temporal distance between the current picture and the reference picture, therefore, the chances of finding the best match with minimum residual are maximum at temporal level-4.

When the prediction Mode is selected as inter, then the CUs can be divided into one two or four PUs. Splitting into four is only allowed at minimum allowed CU size. When the CU is split into four PUs then its divided into four square and equal parts. When its split into two PUs, then there are six possible PUs as shown in Fig. 5. All four depth levels *D*0 to *D*3 with their corresponding PU sizes for N = 32 to 4 are shown. All prediction blocks have one or two motion vectors and corresponding reference indices. In inter picture predicted CUs the partition Modes PART 2N × 2N, PART 2N × N, and PART N × 2N, show that the block is not split, block is split horizontally and vertically into two equal parts respectively. PART N×N is only used at the smallest allowed CU size and splits the CU into 4 equal Parts. There are also

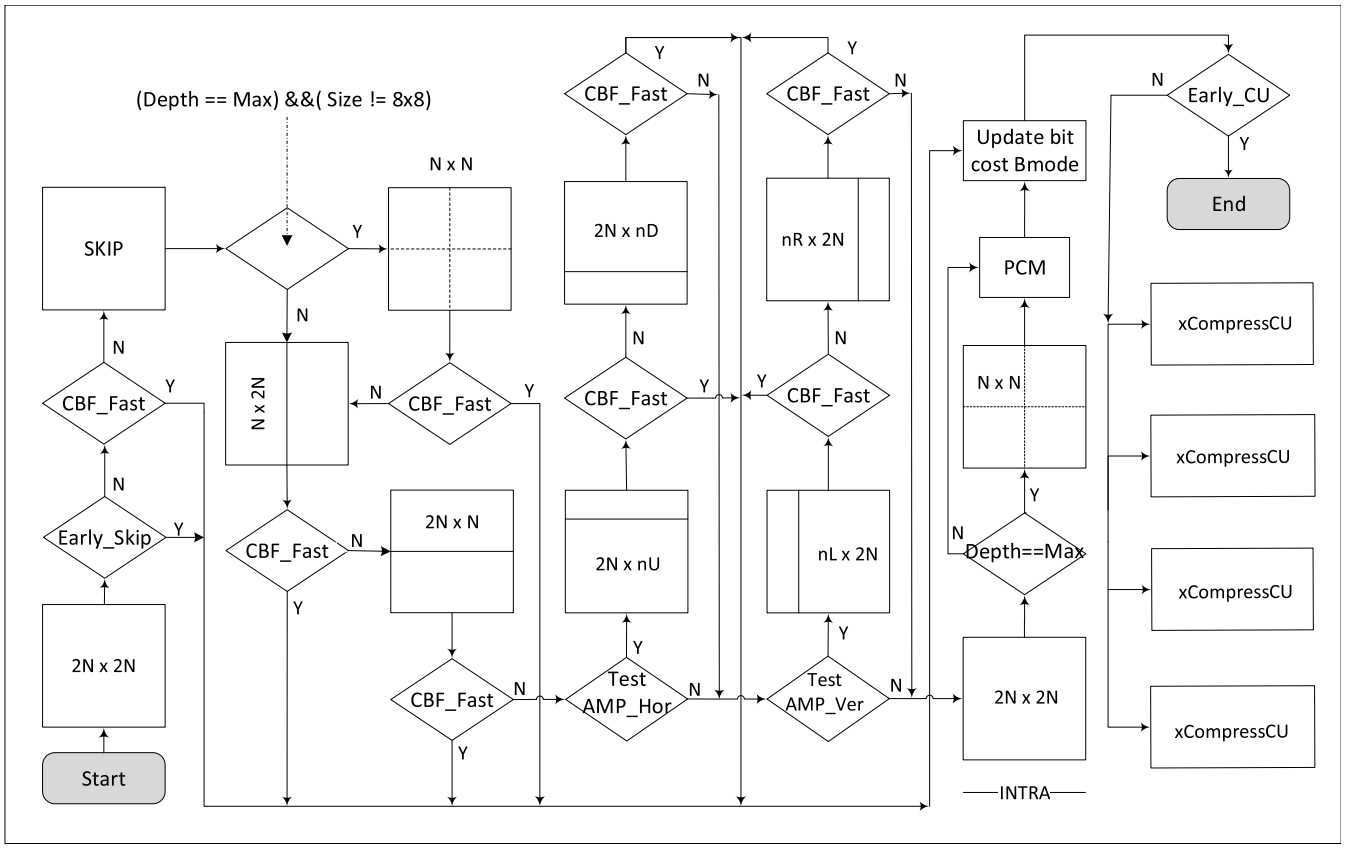


FIGURE 3. CU splitting and PU Mode decision steps, as well as CBF fast, early Skip, and early CU tools for HTM [16.2] [41]. For prediction Modes sizes, please see Fig. 5.

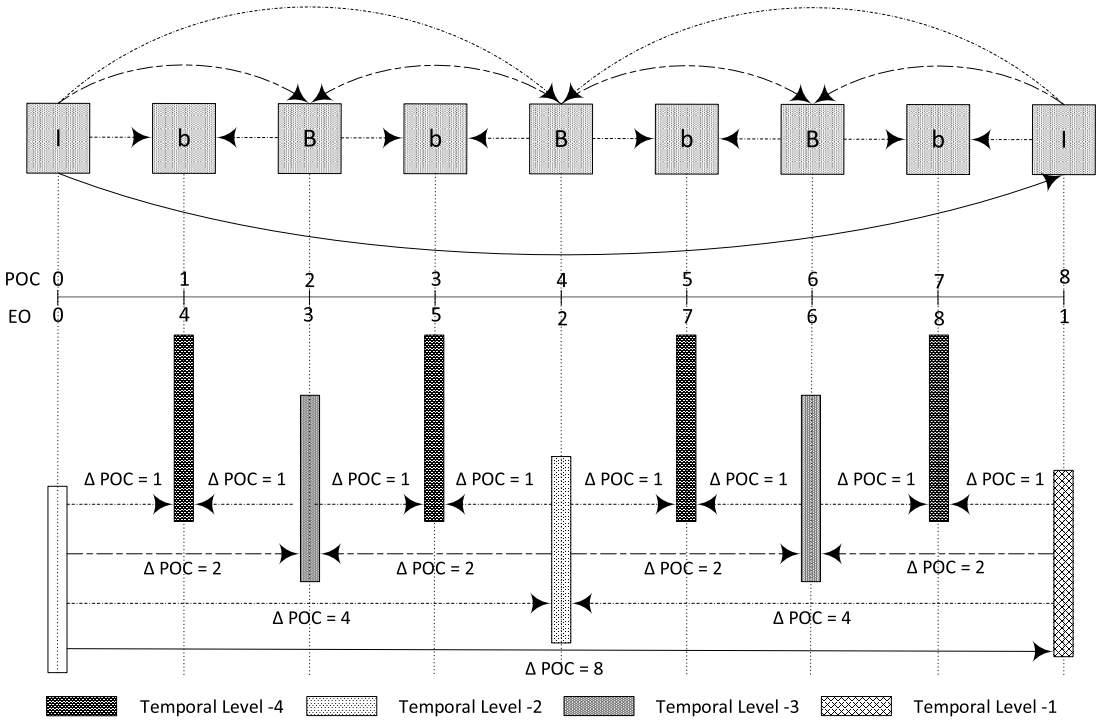


FIGURE 4. Temporal levels (TL): picture order count (POC) and encoding order (EO) for a group of pictures (GOP).

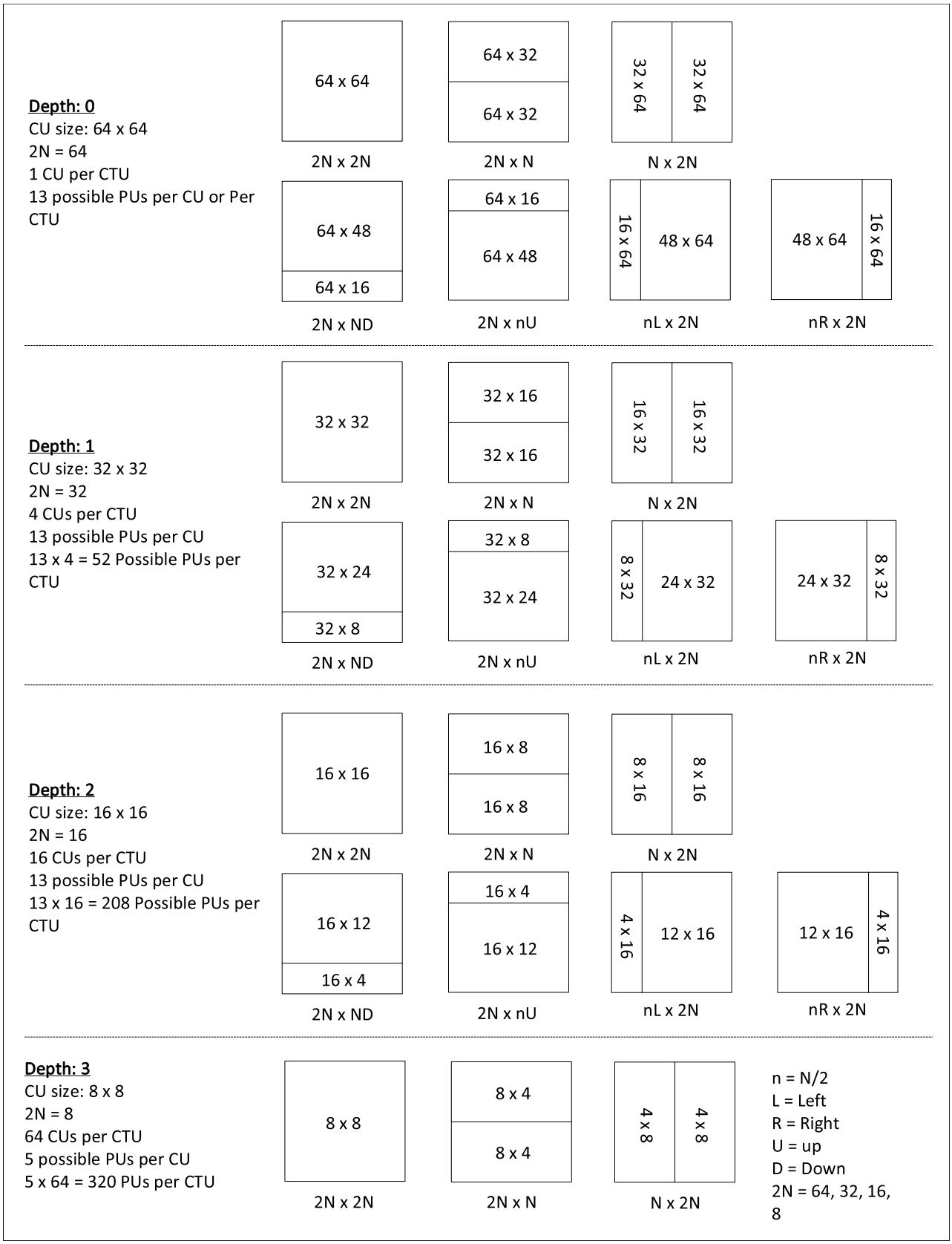


FIGURE 5. The coding tree unit (CTU) partitioning into coding units (CUs), processing order with depth information, and possible prediction units (PUs). (a) CTU partitioning into CUs with processing order, (b) possible PUs for a CU, and (c) corresponding CUs with depth and size information.

TABLE 2. Mode sizes their partition types and mode depth levels.

Modes	Mode Levels	Partition type	Partitions/CU	Partition sizes
SIZE 2N ×2N	0	symmetric motion	2	2N × 2N
SIZE 2N × N	1	symmetric motion	2	2N× N
SIZE N ×2N	2	symmetric motion	2	N × 2N
SIZE N × N	3	symmetric motion	4	N × N
SIZE 2N× nU	4	asymmetric motion	2	2N × (N/2), 2N × (3N/2)
SIZE 2N × nD	5	asymmetric motion	2	2N ×(3N/2), 2N × (N/2)
SIZE nL× 2N	6	asymmetric motion	2	(N/2) × 2N, (3N/2) ×2N
SIZE nR × 2N	7	asymmetric motion	2	(3N/2) ×2N, (N/2)×2N

TABLE 3. The relation of mode (2N × 2N, 2N × N, N × 2N) selection with CU depth levels for different test sequences.

Sequences	Mode level	The percentage of Modes (2N × 2N, 2N × N, and N × 2N) selected by the encoder.									
		Independently					Combined				
		D_0	D_1	D_2	D_3	D_{0-3}	D_0	D_1	D_2	D_3	D_{0-3}
	0	99.2	88.7	86.1	81	88.8					
Balloons	1	0.1	3.5	3.9	3.1	2.7	99.8	97.4	96.1	91.5	96.2
	2	0.5	5.2	6.1	7.4	4.8					
	0	98.1	90.2	86.9	85.7	90.2					
Kendo	1	0.1	1.1	1.1	3.7	1.5	99.5	96.7	95.1	93.1	96.1
	2	1.3	5.4	7.1	3.7	4.4					
	0	99.8	83.7	85.4	85.5	88.6					
Newspaper	1	0.1	3.3	2.5	3.1	2.3	100.0	96.8	94.9	92.5	96.1
	2	0.1	9.8	7	3.9	5.2					
	0	98.6	86.1	78.1	76.8	84.9					
GT_Fly	1	0.2	5.2	5.1	6.5	4.3	99.9	96.0	91.7	91.0	94.7
	2	1.1	4.7	8.5	7.7	5.5					
	0	97.6	85.8	77.3	76.1	84.2					
Poznan_Hall2	1	0.2	4.5	5.5	6	4.1	99.3	93.8	84.0	91.2	92.1
	2	1.5	3.5	1.2	9.1	3.8					
	0	97.8	85.7	73.5	73.3	82.6					
Poznan_Street	1	0.25	6.7	6.1	6.3	4.8	99.3	95.5	90.7	90.0	93.9
	2	1.2	3.1	11.1	10.4	6.5					
	0	97.7	84.6	72.9	72.1	81.8					
Undo_Dancer	1	0.3	7.2	6	6.7	5.1	99.4	95.1	91.3	90.7	94.1
	2	1.4	3.3	12.4	11.9	7.3					
	0	97.9	85.9	76.3	75.6	83.9					
Shark	1	0.4	5.1	5.8	6.1	4.4	99.4	96.7	90.2	89.6	94.0
	2	1.1	5.7	8.1	7.9	5.7					
	0	99.0	87.5	86.1	84.1	89.2					
1024 × 768	1	0.1	2.6	2.5	3.3	2.1	99.8	97.0	95.4	92.4	96.1
	2	0.6	6.8	6.7	5.0	4.8					
	0	97.9	85.6	75.6	74.8	83.5					
1920 × 1088	1	0.3	5.7	5.7	6.3	4.5	99.5	95.4	89.6	90.5	93.7
	2	1.3	4.1	8.3	9.4	5.7					
	0	**98.3**	**86.3**	**79.6**	**78.3**	**85.6**					
Average	**1**	**0.2**	**4.6**	**4.5**	**5.2**	**3.6**	**99.6**	**96.0**	**91.8**	**91.2**	**94.6**
	2	**1.0**	**5.1**	**7.7**	**7.8**	**5.4**					

four asymmetric split sizes, PART 2N × nU, PART 2N × nD, PART nL × 2N, and PART nR × 2N which, split the CU into 2 unequal parts. Different possible Mode sizes used in HEVC are shown in Table. 2. Details about the PUs with respect to their sizes and corresponding CU depth levels can be seen in Fig. 5.

The Mode levels along with the partition type and size of the PU are also shown in Table. 2. This is the numbering given to these Modes in software code of the encoder and we call it Mode depth levels in our proposed method. In the HEVC, the CTU size ranges from 64 ×64 to 16 × 16. The CU size ranges from 64 × 64 to 8 ×8 and PU size from 64 × 64 to 4 × 8/8× 4. Fig. 5 shows all possible PU numbers and sizes with corresponding CU depth levels. The number of possible CUs for a CTU is 85, for depth level $0-3$, $1+4+16+64$ respectively. The number of possible PUs for a CTU is 593, for CU depth levels $0-3$, $13+52+208+320$ respectively.

IV. MOTIVATION

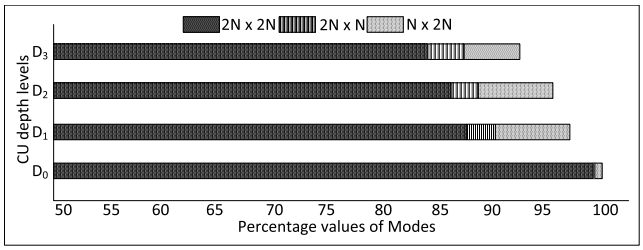
As stated in the previous section, the inter-prediction consumes the majority of the encoding time, therefore, by reducing the complexity of the prediction Mode decisions, complexity of the encoder can be considerably reduced. From the simulation results we find out that the most repeatedly selected Modes are of sizes (2N × 2N, 2N × N, and N × 2N).

In Table. 3 we present the percentage selection of these three Modes, at different CU depth levels for all the test

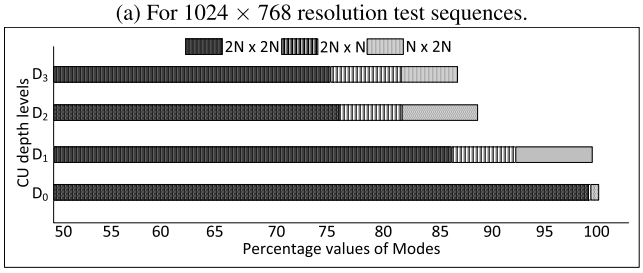
sequences used in this work. Table. 3 not only show the percentage selection of these Modes independently but also describes the relation between them. It can be seen that the Mode 2N × 2N is the most dominantly selected Mode of these three Modes. On average more than 85% of times, Mode 2N × 2N is selected for all test sequences under consideration. This Table. 3 also shows the relation of the these Modes with the CU depth levels. It can be seen that the 2N × 2N Mode selection has the highest probability at the lowest depth level i.e., at D_0 and it decreases as the CU depth level increases. The other two Modes show an increase in the percentage with the increase in the CU depth level.

Mode levels are defined according to the Table 2. For the test sequences of Table 6, and using the average of QP values (25, 30, 35, 40), the percentage selection of these three Modes is shown in Table. 3. First the independent percentage selection of each of these three Modes is shown under the corresponding CU depth level (D_0, D_1, D_2, and D_3) (at which the Modes is selected) separately. Then under the label, D_{0-3} column, we present the percentage selection of each of these three Mode levels for all the depth levels. This gives us the information about relation between the Mode selection probability and the CU depth levels and about the Mode selection probability in general for all CU depth levels.

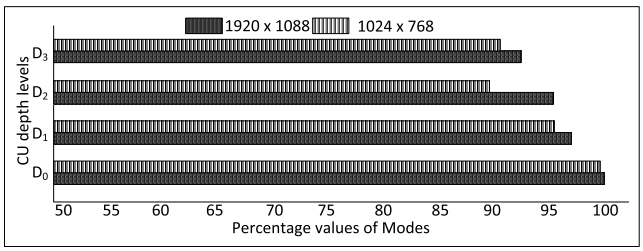
In the columns under the title ''combined'', the collective percentage selection of the three Modes is shown. For the test sequence ''Balloons'' the selection probability of these three Modes for depth D_0 is 99.8 %. Similarly the percentage selection of these three Modes (2N × 2N, 2N × N, and N × 2N) for other CU depth levels are also shown under corresponding depth level D_i. In the right most column, D_{0-3}, the percentage selection of these three CU modes for all the depth levels is shown. For more clarity of the readers, same data is plotted in the Fig. 6a, Fig. 6b separately for 1920 × 1088 and 1024 × 768 resolution video sequences, respectively. Horizontal axis shows the percentage of Mode selection, whereas the vertical axis shows the CU depth levels. In both these graphs Mode 2N × 2N has the highest selection percentage among the three most selected Modes, for all CU depth levels. Specially at CU depth level-0, apart from Mode 2N × 2N, selection of the other two Modes is negligible. The selection percentage of the 2N × 2N Mode decreases as the CU depth level increases, but still it's the most repeatedly selected Mode among these three Modes. Therefore, we can say that, as the CU depth level decreases the chance to select low level Modes increases. As the Fig. 6a and 6b, present the same information for different resolution videos, therefore the effect of resolution can also be seen from the difference between the two graphs. In Fig. 6c, the graph shows the relationship between the selection percentage of three Modes (2N × 2N, 2N × N, N × 2N) and the CU depth levels for 1920× 1088 and 1024 × 768 resolution video sequences. Here it can be seen that these three Modes also show a similar trend with respect to the CU depth level.



(a) For 1024 × 768 resolution test sequences.



(b) For 1920 × 1088 resolution test sequences.



(c) Selection of any of the three most selected Modes.

FIGURE 6. The percentage selection of the three most selected modes (2N × 2N, 2N × N, and N × 2N) and with the CU depth level.

The highest selection percentage of these Modes is at the lowest CU depth level and decreases with the increase in CU depth level. From the results of Table 3 and Fig. 6a, 6b, and 6c, its evident that:

- These three Mode sizes (2N × 2N, 2N × N, and N × 2N) have a high percentage of selection as the best Mode.
- This selection percentage further increases with the decrease in CU depth level.
- It is evident that out of these three Most selected Modes the Mode 2N × 2N, is the most selected Mode, which means that in most of the cases we even do not require to check the other two Modes i.e., 2N × N, and N × 2N.
- We see that, the percentage of Mode 2N × 2N with respect to the other two Modes (2N × N, N × 2N) is higher for low resolution videos.
- We also note that the Percentage of all the three Modes (2N × 2N, 2N × N, N × 2N) with respect to other Modes is higher for low resolution videos, which means that we can get more time reduction in case of low resolution videos.

This gives us motivation that we can reduce the complexity of the encoder, if we can develop a technique through which we can predict these Modes without

TABLE 4. Depth levels of CUs, their corresponding area in pixels, and ratio with respect to the LCU is shown in the table.

CU Depth level	Size in pixels	Ratio with respect to LCU
0	64 × 64	1
1	32 × 32	1/4
2	16 × 16	1/16
3	8 × 8	1/64

going through the complex default process. We need to find out:

- When we need to check only Mode 2N × 2N, and
- When the we also need to check the other two Modes 2N × N, and N × 2N.

Our proposed technique is explained in the next section.

V. PROPOSED EFFICIENT INTER-PREDICTION MODE DECISION (EIPMD) TECHNIQUE

In HEVC, for each 2N × 2N size CU block, both symmetric and asymmetric PU blocks are used. There are four symmetric and four asymmetric blocks used for PUs, for different CU sizes these blocks are known as PU Modes. Depending upon the depth of the CU level each CU is predicted by using these block sizes. The Quad-tree CU splitting mechanism of 64 × 64 block and these prediction Modes for each CU block, further increase the complexity of the encoder. Therefore, optimizing the Mode selection process can reduce the encoding complexity of the encoder considerably. From Section IV we find out that the most probable Modes are 2N × 2N, 2N × N, N × 2N and rest of the Modes are selected very rarely. In low complexity Multi-view video coding the Mode decisions of the base view are used to optimize the Mode decision process of the dependent views [46]–[48]. Since we also intend to optimize the Mode selection process for dependent views, therefore the Mode decisions of the independent view/ base view can also be used for predicting the Modes for dependent views. In our proposed method, we utilize the correlation between different parameters. These parameters and their correlation that we utilized to develop our proposed technique are explained below, one by one.

A. TEMPORAL LEVEL

We discussed in section III, the temporal levels of the pictures with the help of Fig. 4, which shows the temporal levels of a GOP. Due to the availability of temporally adjacent reference pictures at temporal level-4, 95.18% of the CUs are encoded with depth level zero, 2.92% at depth level-1, 1.11% at depth level-2, and 0.78% are encoded at depth level-3 [49]. When taking into account the sizes of the CUs at corresponding depth levels, the percentage area of the picture covered by depth level-0 CUs is 99.15% of the picture area. The relation between CU depth levels, their sizes are the Ratio with the Largest Coding Unit (LCU) is shown in Table 4.

As the temporal level increases, inter-prediction becomes more dominant mode of prediction, due to the availability of temporally closer reference pictures. Reference pictures are at smaller temporal distance, therefore the probability of finding the best matches in these reference pictures increases many folds. At temporal level-4 we have adjacent pictures in the reference picture set of the picture being encoded, so in order to reduce the complexity of the encoder, we can avoid checking for intra-prediction Modes at temporal level-4.

Moving objects in a video, degrade its quality. Both the size and speed of the moving object affect the video quality, and to maintain the same quality of the video, the coding bit-rate needs to be increased accordingly [50]. This is addressed by reducing the block sizes, because smaller blocks in larger number can handle complex motion in a better way than fewer larger blocks [51]. Therefore, we can say that small moving objects are a cause of CU split.

As we know that complex motion in a video requires higher CU depth levels, so theoretically it is same to say that complex motion requires higher Mode levels. Mode levels are shown in Table. 2. Combining both the statements we can say that the Mode level in the dependent view is directly proportional with the CU depth level in the co located area of the base view picture. So we use the CU splitting information of the base view and the Modes selection relation with it, to optimize the Modes selection process of the dependent views.

As the reference pictures are available at less temporal distance at higher temporal levels, therefore the possibility of motion between the pictures is also comparatively less. This means lower CU depth levels or in other words lower Mode levels. We use this information in our proposed method and only check the Mode level-0 at the temporal level-4.

B. DECISION WINDOW

Decision window, is the name we use for the area of the base view which is used to predict the Mode decisions of the dependent views. This area is the neighborhood of the co-located CTU in the base view. In our previous work [49], we used a 192 × 192 pixel square box, as a decision window to decide the CU splitting for the dependent views. We use the same decision window in our proposed method, MODm. The decision window used in proposed method, MODm is shown in Fig. 7. It can be noticed, in the three views of ''Balloons'' test sequence, shown in the Fig. 9, the base view and the dependent views are only horizontally displaced pictures of the same scene. In this case View-1 captures the scene with a slight disparity to the base view towards right and View-2 captures with a slight disparity towards left with respect to the base view.

By using this information we further optimize our decision window by using a small area of 128 × 64 pixels box.

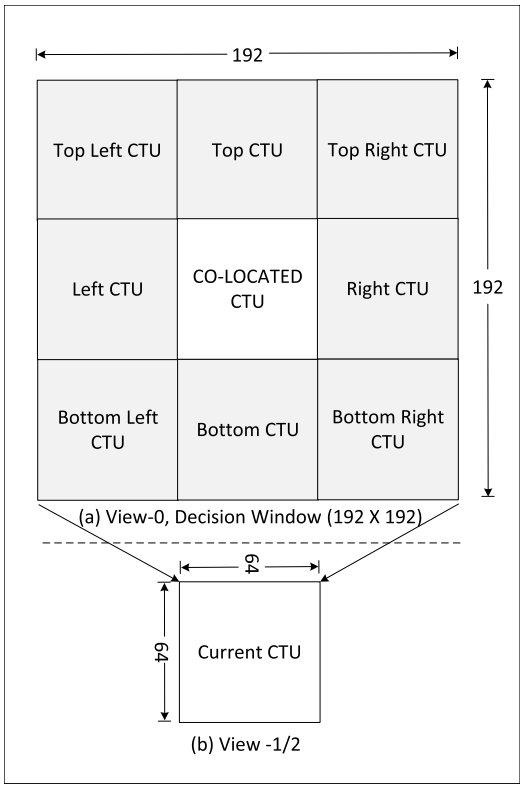


FIGURE 7. Decision window for MODm, (a) neighborhood of the co-located CTU in View–0; (b) the current CTU in the dependent view.

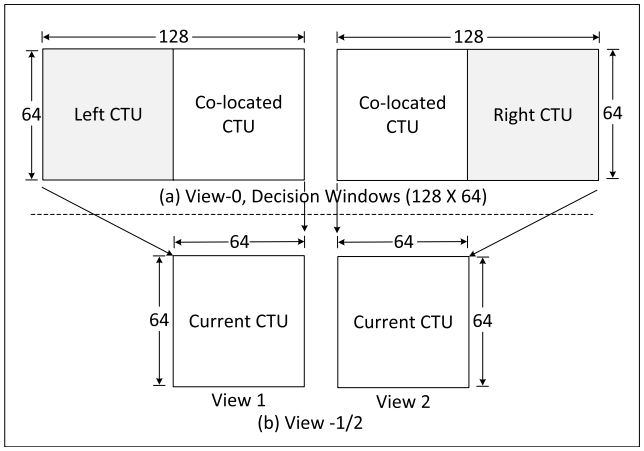


FIGURE 8. Decision window for EIPMD, (a) neighborhood of the co-located CTU in View–0; (b) The current CTU in the dependent view.

For View-1 we use co-located CTU and the CTU towards left and for View-2 we use co-located CTU and the CTU towards right as shown in the Fig. 8. As the disparity between the views is only horizontal hence our concern is only in the region towards right and left of the co-located CTU. This further improves our decision making. Another advantage of using 128 × 64 size decision window as compared with 192 × 192 window is its compactness. The top three and the bottom three CTUs of the 192 × 192 decision window, do not improve the quality or the compression efficiency of the video. At the same time the Mode selection threshold can be increased, due to which encoding time increases slightly. Hence, we remove top and bottom three CTUs of our initial decision window used for MODm. Now we are left with three CTUs. From these three CTUs we find out that due to horizontal disparity between the views, only the left or the right CTU along with collocated CTU gives us useful information. Therefore, we only use two CTUs as our decision window in our proposed EIPMD method, depending upon the view we are encoding. Fig. 8, show the 128 × 64 size decision window that we use for our proposed EIPMD.

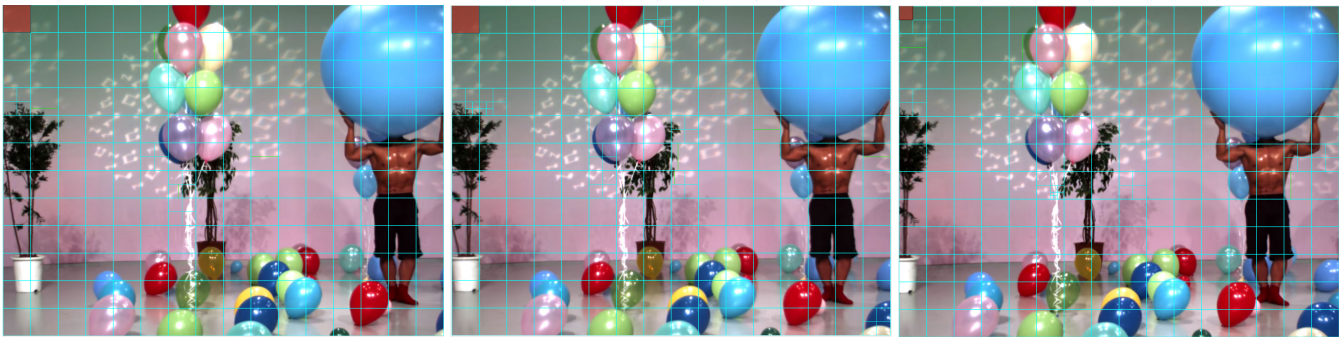
C. BOUNDARY REGION

The region of the picture in the dependent views, where our proposed method is not applicable, we call it boundary region. This is because of the non-availability of the CU splitting information of the base view due to the size of the decision window. For the decision window used in method MODm, the corresponding boundary and non-boundary regions in the dependent views are shown in Fig. 10. Similarly for the decision window used in EIPMD method, the corresponding boundary and non-boundary regions shown in Fig. 11. Size of the decision window restricts the application of our proposed methods on the CUs of non-boundary region CTUs. If we use the 192 × 192 size for decision window than CTUs of the first and last rows and columns of the picture are located in the boundary of the picture, which means our proposed MODm algorithm cannot be applied on two rows and two columns of the CTUs of each picture as shown in Fig. 10. If we use the 128 × 64 rectangular window for decision making than only first or the last column of the CTUs falls into the boundary region, depending upon which view we are encoding, as shown in Fig. 11. So in this case our propose EIPMD algorithm is not applicable only on a single column of CTUs of each picture. Because of being applicable on a larger area of the pictures of the dependent views, the proposed EIPMD technique gives better results.

D. CURRENT AND REFERENCE CU DEPTH

Our proposed method also utilizes the information about the current depth level D_c of the CU being encoded. The algorithm adjusts according to the depth level of the CU that is currently being encoded. This means that for the same CU splitting information from the base view, different Mode decisions are taken for different depth levels of the current CU of the dependent view, that is being encoded. The reasons for this are:

- The correlation between the current CU depth level D_c and the CU depth levels in decision window of the base view as explained earlier.
- If a CU is split into four PUs, the resulting PUs represent square blocks of the same size and this splitting of a CU into four equally sized square PUs i.e., Mode N × N, is theoretically identical to the splitting of the corresponding block into four CUs and coding each of these CUs as a single PU i.e., Mode 2N × 2N [52],



(a) – Dependent view, View-1 (b) – Base view, View-0 (c) – Dependent view, View-2

FIGURE 9. POC 1 of test sequence "Balloons", showing all three views. The CU splitting decisions and the disparity between the views can also be seen for all three views. (a), (b), and (c).

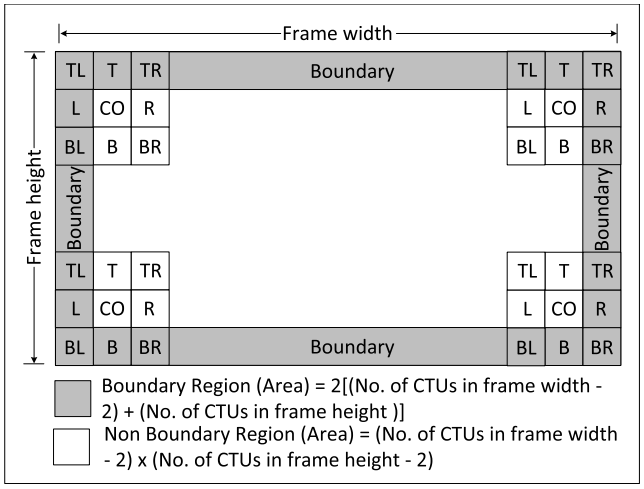


FIGURE 10. Boundary and non-boundary regions in the dependent views, for the decision window used for MODm.

as shown in Fig. 12. In Fig. 12a, a single CU of size 2N × 2N is predicted using four PUs of size N × N, whereas Fig. 12b, shows the same CU is first split into four CUs and then each of these four CUs is predicted using 2N × 2N size single PU.

- Using the above findings, we can say that, the Splitting of a CTU into 4 CUs of size 32 × 32 i.e., at D_1 (after checking Modes at D_0) and checking Mode 2N × 2N for each of the 4 CUs is conceptually similar to checking these three Modes (2N × 2N, 2N × N, N × 2N) without splitting the CU, at D_0.

From our previous work [49], [53], we know that the CU depth of the independent views is directly proportional with the collocated CU depths of the base view. So if we know that the co-located region has CUs of depth level D_0 than it is nearly impossible that the depth of the current CU of the dependent view can have a depth level D_3. Similarly depth level D_2 is also a very rare possibility. Most likely the CU will get the depth level of its co located CU depths. We also use the information from our previous work regarding the relation of CU depth levels between base and dependent views.

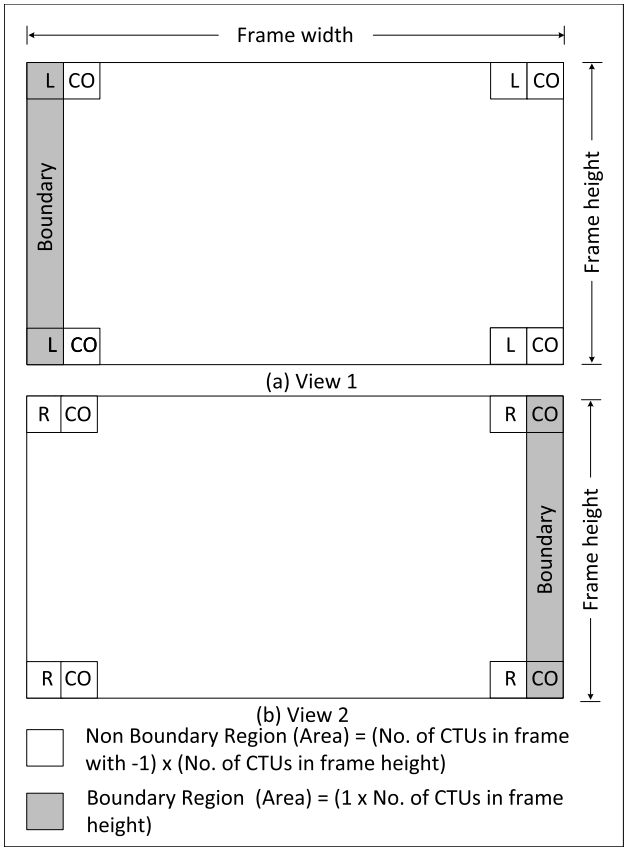


FIGURE 11. Boundary and non-boundary regions in the dependent views, for the decision window used for EIPMD.

In Table. 5, we present the scenarios (which are based on the above findings), that we use in our proposed algorithm1, in which we utilize the correlation between the reference CU depth level D_r and current CU depth level D_c. All possible values of D_c for each values of D_r, are shown. Then corresponding to these D_r values, the possible values of D_c, for each of the three conditions used in the algorithm 1:

- $D_r \leq D_c$
- $D_r - D_c = 1$
- $D_r - D_c = 2$

TABLE 5. Possible CU depth levels of reference and current CUs and the conditions applied in 1.

All possible CU Depths scenarios		Conditions used in the algorithm 1					
Reference CU depth levels	Current CU depth levels	$D_r \leq D_c$		$(D_r - D_c) == 1$		$(D_r - D_c) == 2$	
D_r	D_c	D_r	D_c	D_r	D_c	D_r	D_c
0	0, 1, 2, 3	0	0, 1, 2, 3	0	-	0	-
1	0, 1, 2, 3	1	1, 2, 3	1	0	1	-
2	0, 1, 2, 3	2	2, 3	2	1	2	0
3	0, 1, 2, 3	3	3	3	2	3	1

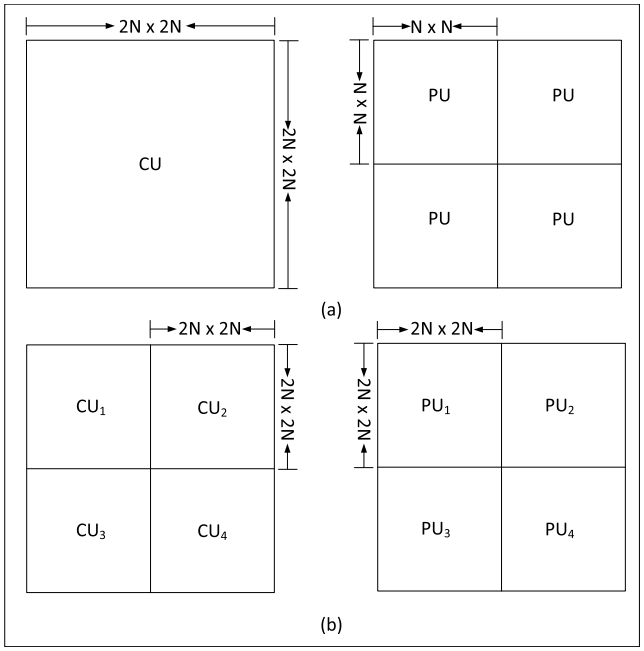


FIGURE 12. (a) A CU at depth level D of size 2N × 2N is predicted into 4 PUs of size N × N; (b) 4 CUs at depth level D+1, and each CU is predicted by single PU of size 2N × 2N.

are shown. For example if $D_r = 3$, then D_c can only have the value of 3 according to the condition $D_r \leq D_c$, can have the value of 2, according to the condition $D_r - D_c = 1$, and can have a value of 1 according to the condition $D_r - D_c = 2$. The condition $D_r \leq D_c$ covers all values of D_c (0, 1, 2, 3) for $D_r = 0$. For $D_r = 1$, it covers $D_c = 1, 2, 3$, whereas, the value $D_c = 1$, is covered by the condition $D_r - D_c = 1$. For $D_r = 2$, the condition $D_r \leq D_c$, covers value of $D_c = 2, 3$, whereas $D_c = 1$ is covered by $D_r - D_c = 1$ and $D_c = 2$ is covered by the condition $D_r - D_c = 2$. Similarly for $D_r = 3$, the condition $D_r \leq D_c$, covers value of $D_c = 3$, whereas $D_c = 2$ is covered by $D_r - D_c = 1$ and $D_c = 1$ is covered by $D_r - D_c = 2$. for the value of $D_c = 0$ default procedure is followed.

E. RD COST

RD cost based fast Mode decision techniques are proposed in the literature for low complexity video coding [34], [46], [54]. In which Mode decisions are taken utilizing the RD costs of the co-located regions of the pictures. RD cost thresholds are calculated based on observation and statistical analysis and then Mode decisions are taken accordingly. We also utilize the RD cost parameter to further enhance our proposed technique. We use the RD cost of the CTUs of the decision window. Complex motion in video demands smaller blocks i.e., higher CU depth levels, which is theoretically same as higher Mode levels. Higher CU depth levels or PU Mode levels means higher RD cost. On the bases of this information we utilize the average RD cost values of the CTUs of the decision window to further improve our proposed Mode decision technique. We calculate the average RD cost ($CostAvg_{Ref}$) for the co-located CTU and CTUs of the decision window. We also calculate average RD cost ($CostAvgDepth_{Ref}$) for each CTU for all depth levels in a picture of the base view. Using these RD cost values we decide between the Modes to be selected.

F. PROPOSED ALGORITHM

After going through the details about the parameters that we used, and their correlations, which we utilized in our proposed method, now we explain the working of our proposed method's Algorithm 1, step wise.

Step 1: First the encoder checks that whether the proposed PU Mode decision algorithms are enabled or not. For this purpose a Boolean variable defined in the encoder. If its enabled the process moves to step 2, otherwise default procedure is followed by the encoder.

Step 2: Next the encoder checks whether the MODm technique is activated or the EIPMD technique is activated. Corresponding details for Decision Window (DW) are acquired for the activated technique.

Step 3: After deciding about the DW, if the current view is view-0 i.e., the base view, then D_r is acquired and saved. Where D_r is the maximum depth level of a CU that appears in DW corresponding to a CU in dependent view. Here we save the vales for D_r, which are used afterwards while encoding the dependent views.

Step 4: As explained earlier about the limitations in the area of the picture, where our proposed methods are applicable, here we check about the boundary condition. If the current CU belongs to the boundary region of the picture than default procedure is used.

Step 5: If the current CU does not belong to the boundary region of the picture then, the proposed algorithms

check for two conditions. First it checks whether the picture belongs to temporal level 4 or not. Then it checks whether the reference CU depth level D_r is less than or equal the current CU's depth level, D_c. If either of these conditions is true, the encoder is restricted to only predict the current dependent view CU using PU Mode size 2N × 2N. If none of these conditions is satisfied, then the process moves to the next step.

Step 6: Algorithm checks, whether the reference depth D_r is one level above the current CU depth D_c. If the conditions turns out to be false the algorithm moves to the next step. If the condition turns out to be true, then the algorithm checks the cost criteria (the average RD cost of the co-located CTU is less than that of the average RD cost calculated for its corresponding depth times a scaling factor, N2 = 0.75). If the cost criteria is satisfied the encoder is restricted to predict the current CU with PU size 2N × 2N, otherwise the encoder is restricted to 2N × 2N, 2N × N, and N × 2N PU sizes.

Step 7: A similar mechanism to step 6 is followed, with a slight difference. Here the algorithm checks, whether the reference depth D_r is two levels above the current CU depth D_c. If the condition turns out to be false default procedure for prediction is followed. If the condition is true, again the cost criteria (the average RD cost of the co-located CTU is less than that of the average RD cost calculated for its corresponding depth times a scaling factor, N2 = 0.75) is checked. If the cost criteria is satisfied, the perdition for current CU is restricted to 2N × 2N, 2N × N, and N × 2N PU sizes, otherwise, default prediction Modes are used.

Algorithm 1: Proposed PU Mode Selection Algorithm

```
if ( Enable PU Mode decision ) then
    if ( MODm ) then
        DW = W_ MODm
    if ( EIPMD ) then
        DW = W_ EIPMD
    if ( Current view == 0 ) then
        D_r = max(CU depths inside DW)
    else if ( !boundary) then
        if ( Temp − Level == 4 ) || ( D_r <= D_c ) then
            Check Mode 2N × 2N
        else if ( (D_r − D_c) == 1 ) then
            if ( CostAvg_Ref ) < N2 × CostAvgDepth_Ref ) then
                Check Modes
                2N × 2N
            else
                Check Modes
                2N × 2N
                2N × N
                N × 2N
        else if ( (D_r − D_c) == 2) then
            if ( CostAvg_Ref ) < N2 × CostAvgDepth_Ref ) then
                Check Modes
                2N × 2N
                2N × N
                N × 2N
            else
                Default MV-HEVC Algorithm.
        else
            Default MV-HEVC Algorithm.
    else
        Default MV-HEVC Algorithm.
else
    Default MV-HEVC Algorithm.
```

VI. EVALUATION PARAMETERS AND EXPERIMENTAL SETUP

In this section we briefly define the parameters that we used for comparing our proposed technique with reference encoder. To give an extra insight into the complexity reduction process, in our proposed work we also present the percentage change in the arithmetic operations along with the standard comparison parameters. The dominant factor in computational complexity is the repetitive execution of arithmetic calculations such as Hadamard (HAD) transform, Sum of Absolute Difference (SAD), and Sum of Squared Error (SSE) [55]. Block matching is a process in which the best match of the block being encoded is searched in a search window both spatially and temporally. All available CU sizes and their possible PU sizes are explored with certain conditions. for the best match difference between the blocks is calculated by (1).

$$Diff(i,j) = BlockA(i,j) - BlockB(i,j) \tag{1}$$

where i and j the pixel location. The SSE and SAD function are defined by (2) and (3) respectively.

$$SSE = \sum_{i,j} Diff(i,j)^2 \tag{2}$$

$$SAD = \sum_{i,j} |Diff(i,j)| \tag{3}$$

During the process of finding the best match, the number of bit required to represent the residual along with splitting details, is also considered. For the overall calculation of Rate Distortion cost, (4) and (5) are utilized.

$$J_{pred,SAD} = SAD + \lambda_{pred} * B_{pred} \tag{4}$$

$$J_{mode} = (SSE_{lu.} + w_{ch.} * SSE_{ch.}) + \lambda_{mode} * B_{mode} \tag{5}$$

where B_{mode} is bit cost for Mode decision and B_{pred} is bit cost for the prediction decision. λ_{pred} and λ_{mode} are Lagrange multipliers. $_{lu}$ and $_{ch}$ stand for luma and chroma, respectively.

TABLE 6. Test sequences used in the analysis and results.

S.No	Test Sequence	Resolution	Input Views	Frames/Pictures
1	Balloons	1024×768	$1-3-5$	300
2	Kendo	1024×768	$1-3-5$	300
3	Newspaper	1024×768	$2-4-6$	300
4	GT_Fly	1920×1088	$9-5-1$	250
5	Poznan_Hall2	1920×1088	$7-6-5$	200
6	Poznan_Street	1920×1088	$5-4-3$	250
7	Undo_Dancer	1920×1088	$1-5-9$	250
8	Shark	1920×1088	$1-5-9$	300

TABLE 7. Experimental setup and encoder configurations.

Hardware	Intel core i7, 2.7 GHz processor, and 32 GB RAM
Operating system	Windows 7
Encoder configuration	3D-HTMEncoder: Version [16.2] based on HMVersion [16.9][Windows][VS 1900][32 bit], Profile: main main multiview-main, CU size /total depth: 64/4, Intra-period: 24, GOP : 8, NumberOfLayers: 3, HadamardME:1, Motion search range: 64, Disp. search range restriction: 1, Vertical disp. search range: 56, BipredSearchRange: 4, HAD: 1, RDQ: 1, RDQTS: 1, MinSearchWindow: 8, RestrictMESampling: 1, ECU: 0, ESD: 0, CFM: 0, ASR: 0, FEN: 0, FDM: 0, TransformSkip: 1, TransformSkipFast: 1, Trans -formSkipLog2MaxSize: 2, Slice: M = 0, SliceSegment: PME: 2, WaveFrontSubstreams: 1, TMVPMode: 1,SignBitHidingFlag: 1.

w_{ch} is the weighting factor for the chroma part of SSE. The arithmetic functions are executed for caparisons to get the best match during the encoding process. The number of time these arithmetic functions are called, also gives an insight into the encoding complexity. Therefore, we also compare our proposed method with the reference encoder on the basis of the percentage decrease in the number of these arithmetic operations. Equations (6-8) are used for the comparison purpose. These equations give us the percentage change in the HAD, SAD, and SSE operations for our proposed technique.

$$\Delta HADs(\%) = \frac{HADs_{(Orig.)} - HAD_{(Prop.)}}{HADs_{(Orig.)}} \times 100\% \quad (6)$$

$$\Delta SAD(\%) = \frac{SAD_{(Orig.)} - SAD_{(Prop.)}}{SAD_{(Orig.)}} \times 100\% \quad (7)$$

$$\Delta SSE(\%) = \frac{SSE_{(Orig.)} - SSE_{(Prop.)}}{SSE_{(Orig.)}} \times 100\% \quad (8)$$

Mostly we use the more generally accepted and the standard evaluation parameters. These parameters, namely percentage change in the bit rate ($\Delta bitrate$ (%)), percentage change in encoding time ($\Delta time$ (%)), and the change in Peak Signal to Noise Ratio (PSNR) (Δ PSNR (dB)), (9)–(11), not only show us the complexity reduction but also give as the information about the compression efficiency and quality of the video.

$$\Delta Bit.R(\%) = \frac{Bit.R_{(Orig.)} - Bit.R_{(Prop.)}}{Bit.R_{(Orig.)}} \times 100\% \quad (9)$$

$$\Delta Time(\%) = \frac{Time_{(Orig.)} - Time_{(Prop.)}}{Time_{(Orig.)}} \times 100\% \quad (10)$$

$$\Delta PSNR(dB) = PSNR_{(Orig.)} - PSNR_{(Prop.)} \quad (11)$$

The reference encoder results are represented by the subscript *"Orig."* and the results of the proposed encoder are represented by the subscript *"Prop."*. Two other widely used parameters Bjøntegaard Delta Bit Rate (BDBR) and Bjøntegaard Delta Peak Signal to Noise Ratio (BDP-SNR) are also used. The Bjøntegaard metric is used to compare the RD curves of two methods, in terms of average PSNR improvement and percent bitrate saving. The BDBR shows the bitrate saving and the BDPSNR shows difference in PSNR [56]. Standard test sequences, shown in Table. 6, are used for simulation purpose with the encoder configurations shown in Table. 7. All the results shown are with respect to the reference encoder HTM [16.2], under common test conditions [57]. Simulation results obtained are discussed in the next section.

VII. RESULTS, ANALYSIS AND COMPARISONS

First we present the results for the percentage change in the values of ΔHAD, ΔSAD, and ΔSSE, calculated using (6–8). The percentage change in these parameters give us an insight into the complexity reduction of the encoder. Table. 8 shows the results for the decision window of size 192×192 as shown in Fig. 7, which is applied on the non-boundary area of the frame/picture as shown in Fig. 10. Similarly Table. 9 shows the results, when decision window of size 128×64 is used as shown in Fig. 8. The boundary and non-boundary area of the picture for this decision window is shown in Fig. 11. These tables show us the percentage reduction in the considered operations, but do not explicitly give us the information about reduction in complexity. Because of the different execution times of the different operations and also of similar operations of different sizes, finding the overall complexity reduction from these tables is difficult. Therefore, we present the results of our proposed EIPMD algorithm in the Table. 10 for decision window size of 192×192 pixel. The results are shown under the title MODm and for decision window of size 128×64 pixel, under the title EIPMD. It can be noted that by optimizing the window size the results have improved. Lower resolution videos perform slightly better than the high resolution videos.

- From Table. 10, it can be seen that our proposed EIPMD technique generally shows similar reduction in complexity for all test sequences.

TABLE 8. For MODm, the percentage reduction in the number of SAD, SSE, and Hadamard (HAD) operations, with decision window of size 192 × 192.

Sequences	ΔSAD								ΔSSE					ΔHADs		
	SAD	4	8	16	12	24	32	64	SSE4	8	16	32	64	HADs	4	8
Balloons	37.2	6.2	12.7	12.9	39.3	36.3	26.5	24.3	9.3	9.3	9.6	9.5	9.1	25.8	36.4	22.2
Kendo	44.7	3.1	12.4	11.3	44.5	45.8	26.7	24.2	9.4	9.5	9.9	10.1	9.8	25.9	36.7	22.4
Newspaper	33.4	5.0	12.7	12.8	33.4	35.4	26.4	24.3	9.2	9.1	9.5	9.5	9.4	25.7	36.3	22.0
GT_Fly	46.3	4.7	11.5	11.9	45.2	44.2	25.1	23.8	9.2	9.1	9.5	9.6	9.0	25.2	35.9	22.1
Poznan_Hall2	39.5	3.1	11.6	12.1	38.9	37.1	24.3	23.9	9.1	9.3	9.6	9.7	9.3	25.8	35.8	22.4
Poznan_Street	44.8	3.6	11.4	12.1	44.5	39.1	22.1	23.7	9.3	8.9	9.4	9.4	9.6	25.1	36.1	22.1
Undo_Dancer	42.1	4.8	12.0	11.8	41.3	40.2	26.5	23.2	8.9	9.0	9.6	9.3	9.2	25.3	36.1	22.1
Shark	48.1	4.6	11.1	11.9	46.4	45.1	25.7	24.1	9.3	9.2	9.4	9.2	9.5	25.6	36.2	22.0
1024 × 768 (avg.)	38.4	4.8	12.6	12.3	39.1	39.2	26.5	24.3	9.3	9.3	9.7	9.7	9.4	25.8	36.5	22.2
1920 × 1088 (avg.)	44.2	4.2	11.5	12.0	43.3	41.1	24.7	23.7	9.2	9.1	9.5	9.4	9.3	25.4	36.0	22.1
Average	**42.0**	**4.4**	**11.9**	**12.1**	**41.7**	**40.4**	**25.4**	**23.9**	**9.2**	**9.2**	**9.6**	**9.5**	**9.4**	**25.6**	**36.2**	**22.2**

TABLE 9. For EIPMD, the percentage reduction in the number of SAD, SSE, and Hadamard (HAD) operations, with decision window of size 128 × 64.

Sequences	ΔSAD								ΔSSE					ΔHADs		
	SAD	4	8	16	12	24	32	64	SSE4	8	16	32	64	HADs	4	8
Balloons	47.4	9.0	14.7	15.3	48.2	48.2	33.7	31.2	12.0	11.9	12.4	12.3	11.7	33.2	46.8	28.5
Kendo	48.3	7.8	14.4	14.2	47.1	49.3	33.8	31.1	12.0	12.1	12.7	12.9	12.2	33.2	46.8	28.6
Newspaper	43.0	8.6	15.0	15.1	42.5	44.9	33.7	31.2	11.8	11.7	12.1	12.2	12.1	33.1	46.7	28.3
GT_Fly	42.5	8.5	14.6	14.5	42.5	43.5	33.5	30.9	11.7	11.7	11.8	11.9	11.3	33.0	46.6	28.1
Poznan_Hall2	42.9	8.6	14.7	14.6	43.2	44.1	33.4	30.8	11.7	11.6	11.7	11.8	11.2	33.1	46.5	28.2
Poznan_Street	43.9	8.1	14.5	14.3	44.5	43.9	33.7	31.0	11.8	11.7	12.1	12.1	11.4	32.5	46.8	28.3
Undo_Dancer	44.1	8.7	14.6	14.4	43.7	45.7	33.6	30.7	12.1	11.5	11.9	12.2	11.5	32.7	46.7	28.1
Shark	45.2	8.4	14.5	14.5	46.5	47.1	33.2	31.1	12.0	11.8	12.1	12.3	12.1	32.8	46.8	28.4
1024 × 768 (avg.)	46.2	8.5	14.7	14.9	45.9	47.5	33.7	31.2	12.0	11.9	12.4	12.5	12.0	33.1	46.8	28.4
1920 × 1088 (avg.)	43.7	8.5	14.6	14.5	44.1	44.9	33.5	30.9	11.9	11.7	11.9	12.1	11.5	32.8	46.7	28.2
Average	**44.7**	**8.5**	**14.6**	**14.6**	**44.8**	**45.8**	**33.6**	**31.0**	**11.9**	**11.8**	**12.1**	**12.2**	**11.7**	**32.9**	**46.7**	**28.3**

- It can be observed that our proposed method produces better results for low resolution videos. The reason for this is that in higher resolution videos the same object of the video is represented by more pixels when compared with low resolution video, which means that there is more chance for higher Mode levels in low resolution video than in higher resolution videos.
- Results for two different window sizes, used by our technique show that the smaller and optimized window produces better results than the larger window size. One reason for this is the fact that the larger window size has more probability of having higher CU depth level than a smaller window size. As our algorithm uses the CU depth information for its decision hence it affects its results. Second reason for better results is the fact that with smaller window size our techniques can be applied on more area of the picture.
- Test sequences with different video content are used. Generally the results are similar and no significant variation in results occurred due to content of the video.

Finally, we compare our proposed EIPMD method with the relative state-of-the art methods, named as Perceptual Distortion Threshold Model (PDTM) [35] and a Fast Mode Decision method (FMD) proposed in [34]. Table. 11 shows the comparison results. All the standard test sequences of Table 6 are used. Average value of the QP values 25, 30, 35, and 40 is used. Comparison is done on the basis of the standard evaluation parameters i.e., percentage change in Bitrate, change in PSNR, percentage Reduction in time (9, 10, 11), BDBR, and BDPSNR.

- Our proposed technique is consistent in complexity reduction in terms of video resolutions and video content.
- On average we decreased the encoding time of the encoder by 66.56%, with Δ Bitrate (%) of −0.948, the Δ PSNR of 0.073.
- The authors in [34], [35], have used ΔTime and BDBR for the sake of comparison. We also used parameters.
- For all test sequences of Table. 6, our proposed technique out performs the [35] and [34], in terms of complexity reduction i.e., the encoding time, with a slight increase in BDBR.
- Similar to our proposed EIPMD method, the method proposed in [35] is also designed for dependent view coding of MV-HEVC. In which edge strength for different blocks is calculated using 3D-Sobel operators. On the basis of these edge strengths the image is divided into two regions. SSE for CUs is estimated using distortion quantization model. Then threshold values are calculated for these regions and Mode decisions are made by comparing it with SSE. This technique uses a novel approach but its repeated calculations also add time to the encoding process. Whereas the FMD [34] technique is originally proposed for HEVC. This technique is applied to MV-HEVC by [35] for the sake of comparison.

TABLE 10. Comparisons between two versions of our proposed method, with different window sizes (MODm vs. EIPMD).

Sequences	QP	Δ Bitrate (%)		ΔPSNR		ΔTime %		BDBR		BDPSNR	
		MODm	EIPMD	MODm	EIPMD	MODm	EIPMD	MODm	EIPMD	MODm	EIPMD
Balloons	25	-1.33	-1.16	0.02	0.03	54.15	67.63	1.97	1.93	-0.08	-0.08
	30	-2.09	-1.89	0.00	0.00	53.84	67.07				
	35	-2.54	-2.01	-0.02	-0.01	52.86	66.92				
	40	-0.08	-1.63	0.05	0.03	53.95	67.08				
Kendo	25	-0.89	-0.83	0.06	0.06	53.44	66.85	2.97	3.28	-0.11	-0.12
	30	-1.47	-1.34	0.06	0.06	53.76	66.88				
	35	-1.62	-1.09	0.06	0.09	53.42	66.90				
	40	1.36	-0.04	0.14	0.16	54.03	66.99				
Newspaper	25	-1.74	-1.79	0.03	0.03	53.48	66.52	2.73	2.56	-0.09	-0.08
	30	-2.92	-2.49	-0.01	0.00	52.60	65.99				
	35	-4.15	-2.51	-0.05	-0.01	52.27	65.84				
	40	-2.37	-1.67	0.00	0.03	52.19	65.98				
GT_Fly	25	-0.38	-0.23	0.09	0.09	54.15	67.63	2.71	2.30	-0.10	-0.09
	30	-1.03	-0.88	0.00	-0.01	53.84	67.07				
	35	-2.89	-2.24	0.06	0.07	52.86	66.92				
	40	-0.29	-0.32	0.03	0.01	53.95	67.08				
Pozna_Hall2	25	-0.61	-0.57	0.20	0.20	53.44	66.85	3.49	2.89	-0.16	-0.14
	30	-0.61	0.61	0.20	0.21	53.76	66.88				
	35	-1.11	-0.33	0.01	0.02	53.42	66.90				
	40	-1.02	-1.02	0.10	0.09	54.03	66.99				
Poznan_Street	25	-0.53	-0.30	0.12	0.12	53.48	66.52	4.87	3.20	-0.18	-0.12
	30	-1.06	-0.78	0.09	0.09	52.60	65.99				
	35	-1.28	-0.70	0.20	0.10	52.27	65.84				
	40	-2.35	-2.35	0.10	0.00	52.19	65.98				
Undo_Dancer	25	-0.41	-0.14	0.10	0.10	53.48	66.52	4.53	3.64	-0.17	-0.14
	30	-0.54	-0.50	0.20	0.19	52.60	65.99				
	35	-0.98	0.00	0.10	0.09	52.27	65.84				
	40	-0.90	0.69	0.10	0.10	52.19	65.98				
Shark	25	-0.64	-0.32	0.20	0.20	53.48	66.52	4.50	1.62	-0.17	-0.08
	30	-0.85	-0.17	0.10	0.08	52.60	65.99				
	35	-1.54	-0.77	0.10	-0.10	52.27	65.84				
	40	-1.41	-1.55	0.20	0.20	52.19	65.98				
1024 × 768 (avg.)	25	-1.28	-1.03	0.05	0.04	53.69	67.00	2.56	2.59	-0.09	-0.09
	30	-0.74	-1.09	0.07	0.02	53.40	66.64				
	35	-1.97	-1.63	0.02	0.02	52.85	66.55				
	40	-0.55	-1.39	0.06	0.08	53.39	66.68				
1920 × 1088 (avg.)	25	-0.52	-0.31	0.14	0.14	53.61	66.81	4.02	2.73	-0.16	-0.11
	30	-0.82	-0.34	0.12	0.11	53.08	66.38				
	35	-1.56	-0.81	0.09	0.04	52.62	66.27				
	40	-1.19	-0.91	0.11	0.08	52.91	66.40				
Average		**-1.26**	**-0.95**	**0.08**	**0.07**	**53.16**	**66.56**	**3.47**	**2.68**	**-0.13**	**-0.11**

TABLE 11. Comparisons among state of the art methods, [34], [35], and our proposed method, EIPMD.

Sequences	Δ Time %			BDBR		
	Jung FMD [34]	Jiang PDTM [35]	Proposed [EIPMD]	Jung FMD [34]	Jiang PDTM [35]	Proposed [EIPMD]
Balloons	29.30	54.30	67.17	-0.20	1.00	1.93
Kendo	28.70	54.50	66.90	-0.40	2.40	3.28
Newspaper	34.10	54.20	66.08	0.00	1.40	2.56
GT_Fly	33.40	53.70	67.17	0.30	1.30	2.30
Pozna_Hall2	40.00	49.00	66.90	-0.30	2.60	2.89
Poznan_Street	53.80	51.30	66.08	0.10	2.10	3.20
Undo_Dancer	22.20	53.20	66.08	-0.20	3.70	3.64
Shark	22.50	53.40	66.08	-0.10	0.50	1.62
1024 x 768 (avg.)	30.70	54.33	66.72	-0.20	1.60	2.59
1920 x 1088 (avg.)	34.38	52.12	66.46	-0.04	2.04	2.73
Average	**33.00**	**52.95**	**66.56**	**-0.10**	**1.88**	**2.68**

- For test sequence Poznan_Hall2, it can be observed that the performance of method in [35] is reduced. Similar pattern can be seen for sequence Poznan_Street. Whereas for [34], its performance for test sequences

Poznan_Hall2 and Poznan_Street is increased quite a bit. In our proposed EIPMD technique no such video content dependent performance degradation is noticed.
- In terms of BDBR [34] is the best performer followed by [35], and the our proposed technique.

VIII. CONCLUSION AND FUTURE WORK

We propose an Efficient Inter-Prediction Mode Decision (EIPMD) technique to decrease the complexity of MV-HEVC encoder. We present a novel approach for inter-prediction Mode decisions of the dependent views, by utilizing the CU depth level information of the co-located area (we call it the decision window) of the base view. We find that the views are only horizontally displaced and the disparity between the base view and dependent views is only in the horizontal direction. We use this information to further optimize our technique by reducing the size of our decision window. We also utilize the correlation between the Temporal levels of the pictures and the Mode levels. At temporal level-4 adjacent pictures are available for prediction, in the reference picture list. Hence we utilize this to further optimize the Mode decisions in temporal level-4 pictures, by restricting the Mode decision and also by avoiding the intra-prediction. We also enhance the performance of our technique by utilizing the average RD cost of the CTUs of the decision window. We also utilize the correlation between the current CU depth and Mode decisions to further enhance our proposed technique. We define the evaluation parameters and simulate our proposed technique. From the results in Table. 10, it can be seen that our technique shows similar results for all the test sequences. As compared with the reference encoder HTM [16.2], we decreased the encoding time by 66.56% on average, with the Δ Bitrate (%) of -0.95, the Δ PSNR of 0.07, BDBR of 2.679, and BDPSNR of -0.105.

In this work, we only used CU splitting and RD cost information of the base view. In future the Mode selection process can be further improved by utilizing the CU splitting information of the temporally co-located CTUs. Moreover, the search range in the dependent views can also be optimized by utilizing the CU splitting information of the base view, to reduce the complexity of the encoder.

REFERENCES

[1] *VNI Cisco Visual Networking Index (VNI)*. Accessed: Mar. 21, 2021. [Online]. Available: https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2022_Forecast_Highlights.pdf
[2] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Sep. 2012.
[3] *High Efficiency Video Coding*, document ITU-T H.265 and ISO/IEC 23008-2, Jan. 2013.
[4] *Advanced Video Coding for Generic Audiovisual Services*, document ITU-T Recommendation H. 264, 2003.
[5] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards-including high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.
[6] F. Bossen, B. Bross, K. Sühring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1685–1696, Dec. 2012.
[7] H. Urey, K. V. Chellappan, E. Erden, and P. Surman, "State of the art in stereoscopic and autostereoscopic displays," *Proc. IEEE*, vol. 99, no. 4, pp. 540–555, Apr. 2011.
[8] P. Surman, K. Hopf, I. Sexton, W. K. Lee, and R. Bates, "A roadmap for autostereoscopic multi-viewer domestic TV displays," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2006, pp. 1693–1696.
[9] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard," *Proc. IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.
[10] M. Tanimoto, "Overview of free viewpoint television," *Signal Process., Image Commun.*, vol. 21, no. 6, pp. 454–461, Jul. 2006.
[11] C.-C. Lee, A. Tabatabai, and K. Tashiro, "Free viewpoint video (FVV) survey and future research direction," *APSIPA Trans. Signal Inf. Process.*, vol. 4, p. e15, Oct. 2015.
[12] K. J. Oh, S. Yea, A. Vetro, and Y.-S. Ho, "Virtual view synthesis method and self-evaluation metrics for free viewpoint television and 3D video," *Int. J. Imag. Syst. Techn.*, vol. 20, no. 4, pp. 378–390, Dec. 2010.
[13] M. M. Hannuksela, Y. Yan, X. Huang, and H. Li, "Overview of the multiview high efficiency video coding (MV-HEVC) standard," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 2154–2158.
[14] S. Cho and M. Kim, "Fast CU splitting and pruning for suboptimal CU partitioning in HEVC intra coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 9, pp. 1555–1564, Sep. 2013.
[15] Y. Zhang, S. Kwong, G. Zhang, Z. Pan, H. Yuan, and G. Jiang, "Low complexity HEVC INTRA coding for high-quality mobile video communication," *IEEE Trans. Ind. Informat.*, vol. 11, no. 6, pp. 1492–1504, Dec. 2015.
[16] K. Choi, S. Park, and E. Jang, *Coding Tree Pruning Based CU Early Termination*, document JCTVC-F092, JCT-VC, Torino, Italy, 2011.
[17] K. Won, H. Lee, B. Jeon, J. Yang, and J. Kim, *Early Skip Detection for HEVC*, document JCTVC-G543, JCT-VC document, 2011.
[18] R.-H. Gweon and Y.-L. Lee, "Early termination of cu encoding to reduce HEVC complexity," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. 95, no. 7, pp. 1215–1218, 2012.
[19] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 465–470, Feb. 2012.
[20] H.-S. Kim and R.-H. Park, "Fast CU partitioning algorithm for HEVC using an online-learning-based Bayesian decision rule," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 130–138, Jan. 2015.
[21] J. Xiong, H. Li, Q. Wu, and F. Meng, "A fast HEVC inter CU selection method based on pyramid motion divergence," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 559–564, Feb. 2013.
[22] H.-M. Yoo and J.-W. Suh, "Fast coding unit decision based on skipping of inter and intra prediction units," *Electron. Lett.*, vol. 50, no. 10, pp. 750–752, May 2014.
[23] J. Xiong, H. Li, F. Meng, S. Zhu, Q. Wu, and B. Zeng, "MRF-based fast HEVC inter CU decision with the variance of absolute differences," *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2141–2153, Dec. 2014.
[24] J. Lee, S. Kim, K. Lim, and S. Lee, "A fast CU size decision algorithm for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 411–421, Mar. 2014.
[25] J. Xiong, H. Li, F. Meng, Q. Wu, and K. N. Ngan, "Fast HEVC inter CU decision based on latent SAD estimation," *IEEE Trans. Multimedia*, vol. 17, no. 12, pp. 2147–2159, Dec. 2015.
[26] C. Rhee, K. Lee, T. Kim, and H.-J. Lee, "A survey of fast mode decision algorithms for inter-prediction and their applications to high efficiency video coding," *IEEE Trans. Consum. Electron.*, vol. 58, no. 4, pp. 1375–1383, Nov. 2012.
[27] L. Shen, Z. Zhang, and Z. Liu, "Adaptive inter-mode decision for HEVC jointly utilizing inter-level and spatiotemporal correlations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 10, pp. 1709–1722, Oct. 2014.
[28] H. Lee, H. J. Shim, Y. Park, and B. Jeon, "Early skip mode decision for HEVC encoder with emphasis on coding quality," *IEEE Trans. Broadcasting*, vol. 61, no. 3, pp. 388–397, Sep. 2015.
[29] X. Huang, Q. Zhang, X. Zhao, W. Zhang, Y. Zhang, and Y. Gan, "Fast inter-prediction mode decision algorithm for HEVC," *Signal, Image Video Process.*, vol. 11, no. 1, pp. 33–40, Jan. 2017.
[30] Z. Pan, S. Kwong, M.-T. Sun, and J. Lei, "Early MERGE mode decision based on motion estimation and hierarchical depth correlation for HEVC," *IEEE Trans. Broadcast.*, vol. 60, no. 2, pp. 405–412, Jun. 2014.
[31] Y.-X. Song and K.-B. Jia, "Early merge mode decision for texture coding in 3D-HEVC," *J. Vis. Commun. Image Represent.*, vol. 33, pp. 60–68, Nov. 2015.

[32] Z. Pan, Y. Zhang, J. Lei, L. Xu, and X. Sun, "Early DIRECT mode decision based on all-zero block and rate distortion cost for multiview video coding," *IET Image Process.*, vol. 10, no. 1, pp. 9–15, Jan. 2016.

[33] H. R. Tohidypour, M. T. Pourazad, and P. Nasiopoulos, "A low complexity mode decision approach for HEVC-based 3D video coding using a Bayesian method," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 895–899.

[34] S.-H. Jung and H. W. Park, "A fast mode decision method in HEVC using adaptive ordering of modes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 10, pp. 1846–1858, Oct. 2015.

[35] G. Jiang, B. Du, S. Fang, M. Yu, F. Shao, Z. Peng, and F. Chen, "Fast inter-frame prediction in multi-view video coding based on perceptual distortion threshold model," *Signal Process., Image Commun.*, vol. 70, pp. 199–209, Feb. 2019.

[36] G. Tech, Y. Chen, K. Müller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, Jan. 2015.

[37] I.-K. Kim, J. Min, T. Lee, W.-J. Han, and J. Park, "Block partitioning structure in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1697–1706, Dec. 2012.

[38] H. Samet, "The quadtree and related hierarchical data structures," *ACM Comput. Surv.*, vol. 16, no. 2, pp. 187–260, Jun. 1984.

[39] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," *IEEE Trans. Image Process.*, vol. 3, no. 3, pp. 327–331, May 1994.

[40] J. Zhang, M. O. Ahmad, and M. N. S. Swamy, "Quadtree structured region-wise motion compensation for video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 5, pp. 808–822, Aug. 1999.

[41] R. Chris, B. Benjamin, N. Matteo, S. Karl, and S. Garry, *High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Improved Encoder Description Update 8, JCTVC-AA1002*, document JCT-VC ITU-T SG16 WP3 ISO/IEC JTC1/SC29/WG11, 2017.

[42] G. Correa, P. Assuncao, L. Agostini, and L. A. da Silva Cruz, "Performance and computational complexity assessment of high-efficiency video encoders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1899–1909, Dec. 2012.

[43] F. Sampaio, S. Bampi, M. Grellert, L. Agostini, and J. Mattos, "Motion vectors merging: Low complexity prediction unit decision heuristic for the inter-prediction of HEVC encoders," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2012, pp. 657–662.

[44] M. Grellert, M. Shafique, M. U. K. Khan, L. Agostini, J. C. Mattos, and J. Henkel, "An adaptive workload management scheme for HEVC encoding," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 1850–1854.

[45] J.-L. Lin, Y.-W. Chen, Y.-W. Huang, and S.-M. Lei, "Motion vector coding in the HEVC standard," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 957–968, Dec. 2013.

[46] J. Lei, J. Sun, Z. Pan, S. Kwong, J. Duan, and C. Hou, "Fast mode decision using inter-view and inter-component correlations for multiview depth video coding," *IEEE Trans. Ind. Informat.*, vol. 11, no. 4, pp. 978–986, Aug. 2015.

[47] L. Shen, Z. Liu, T. Yan, Z. Zhang, and P. An, "Early SKIP mode decision for MVC using inter-view correlation," *Signal Process., Image Commun.*, vol. 25, no. 2, pp. 88–93, Feb. 2010.

[48] L.-Q. Shen, Z.-Y. Zhang, and G.-B. Yang, "Adaptive early termination of fast mode decision in MVC based on inter-view correlation," *J. Shanghai Univ. English Ed.*, vol. 15, no. 4, pp. 321–324, Aug. 2011.

[49] S. Khan, N. Muhammad, S. Farwa, T. Saba, and Z. Mahmood, "Early CU depth decision and reference picture selection for low complexity MV-HEVC," *Symmetry*, vol. 11, no. 4, p. 454, Apr. 2019.

[50] T.-M. Pan, K.-C. Fan, and Y.-K. Wang, "Object-based approach for adaptive source coding of surveillance video," *Appl. Sci.*, vol. 9, no. 10, p. 2003, May 2019.

[51] J. Ribas-Corbera and D. L. Neuhoff, "Optimal block size for block-based motion-compensated video coders," *Proc. SPIE*, vol. 3024, pp. 1132–1143, Jan. 1997.

[52] H. Schwarz, T. Schierl, and D. Marpe, "Block structures and parallelism features in HEVC," in *High Efficiency Video Coding (HEVC)* (Integrated Circuits and Systems), V. Sze, M. Budagavi, and G. Sullivan, Eds. Cham, Switzerland: Springer, 2014. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-06895-4_3, doi: 10.1007/978-3-319-06895-4_3.

[53] S. N. Khan and S. Khattak, "Early decision of CU splitting, using base view information, for low complexity MV-HEVC," in *Proc. Int. Multi-topic Conf. (INMIC)*, Nov. 2017, pp. 1–6.

[54] W. Li, B. Zhou, and X. Huang, "Fast inter mode decision based on RD costs and frequencies of modes," *Int. J. Distrib. Sensor Netw.*, vol. 5, no. 1, p. 18, 2009.

[55] F. Saab, I. H. Elhajj, A. Kayssi, and A. Chehab, "Profiling of HEVC encoder," *Electron. Lett.*, vol. 50, no. 15, pp. 1061–1063, Jul. 2014.

[56] B. Gisle, *Calculation of Average PSNR Differences Between RD-Curves*, document ITU-T VCEG-M33, Apr. 2001.

[57] M. Karsten and V. Anthony, *Common Test Conditions of 3DV Core Experiments*, document JCT3V-G1100, Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V), San Jose, CA, USA, Jan. 2014.

SHAHID NAWAZ KHAN received the B.Sc. degree in electrical engineering from the University of Engineering and Technology, Peshawar, Pakistan, in 2003, and the M.Sc. degree in electrical engineering from Linköping University, Sweden, in 2010. Currently, he is pursuing the Ph.D. degree in electrical engineering with COMSATS University Islamabad (CUI), Abbottabad Campus, Pakistan. His current research interests include low complexity techniques for multi-view high efficiency video coding. He was a recipient of the Scholarship for M.S. studies from CUI, from 2008 to 2010.

KHURRAM KHAN received the B.S. degree in computer engineering from COMSATS University Pakistan, in 2007, and the M.S. and Ph.D. degrees in electronic and communication engineering from Hanyang University, South Korea, in 2010 and 2019, respectively. Currently, he is with the Department of Avionics Engineering, Air University, Islamabad, Pakistan. His research interests include computer vision, pattern recognition, and machine/deep learning.

NAZEER MUHAMMAD received the Ph.D. degree in applied mathematics from Hanyang University, South Korea, in 2015. Currently, he is with the Department of IT and Computer Science, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Haripur, Pakistan. His research interests include digital signal watermarking, data hiding, digital image denoising, digital holography, OFDM, and information theory. In 2010, he received the prestigious Pakistan Government Higher Education Commission (HEC) Scholarship Award for his M.S. and Ph.D. studies.

ZAHID MAHMOOD received the B.S. degree in electrical and computer engineering from COMSATS University Islamabad (CUI), Abbottabad Campus, Pakistan, in 2007, the M.S. degree in electrical engineering from Hanyang University, South Korea, in 2011, and the Ph.D. degree in electrical engineering from North Dakota State University, USA, in 2015. His work has appeared in over 45 publications. His research interests include object detection, pattern recognition, image enhancement, and digital image/video processing. He was a recipient of the Higher Education Commission, Government of Pakistan Scholarship Award for his M.S. and Ph.D. studies.